

REMARKS/ARGUMENTS

The claims have been amended as set forth above to further clarify features of the claims. No new matter has been added. Applicants respectfully request reconsideration.

I. Examiner Interview Dated November 20, 2007

An interview was held on November 20, 2007. An agreement as to allowability was not reached. Applicants believe that an agreement was reached that the current changes push prosecution over the cited references.

II. Rejection of the Claims Under 35 U.S.C. 103(a)

Claims 1-5, 7-12, and 14-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over "XML Namespaces by Example, 1999" authored by Bray (hereinafter "Bray") in view of U.S. Publication No. 2002/0065110 published to Enns et al (hereinafter "Enns"). Claims 6 and 13 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bray in view of Enns and further in view of U.S. Publication No. 2004/0103199 published to Chao et al. (hereinafter "Chao"). Applicants respectfully disagree with the rejection.

Independent claim 1 includes the following combination of features that is not taught or otherwise suggested by the cited references:

analyzing a tag in the ML document;

referencing a definition file location attribute in the ML document, wherein the definition file location attribute is identified by the tag;

retrieving a definition file from a location identified by the definition file location attribute, wherein the definition file includes a list of common language runtime namespaces, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace;

referencing **a common language runtime namespace** related to the tag within the definition file to determine the **common language runtime class** associated with the tag; and

locating the **common language runtime class** in an assembly such that the tag is mapped to the **common language runtime class**.

Bray does not teach or otherwise suggest the above combination of features. Bray teaches an example of a namespace and explains the general purpose of having a namespace. On page one of Bray, Bray teaches defining a prefix for a name space. The prefix is used as shorthand for the namespace which is defined at the top of the XML scenario. Bray teaches that namespaces are used to help computer software designers to separate content into namespaces without collision. Bray teaches that attributes can have namespaces too. Enns teaches accessing a DLL to obtain a Plug-in. *Enns* at [0067]. Enns teaches that an application accesses the DLL to obtain functions by creating a link to the DLL. Enns further teaches that "[t]he file attribute associated with the 'plugin' specifies a DLL file that includes the plug-in and the name attribute specifies the name of the plug-in within the DLL." *Enns* at [0068]

Neither Bray nor Enns teaches or otherwise suggests "retrieving a definition file from a location identified by the definition file location attribute, wherein the definition file includes a list of common language runtime namespaces, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace," "referencing a common language runtime namespace related to the tag within the definition file to determine the common language runtime class associated with the tag," and "locating the common language runtime class in an assembly such that the tag is mapped to the common language runtime class." In fact, neither Bray nor Enns even mention a common language runtime namespace, a common language runtime class and/or utilizing a tag to reference the common language runtime namespace to determine the common language runtime class. Also, neither reference teaches mapping the tag to the common language runtime class. Accordingly, applicants assert that claim 1 is allowable.

Independent claim 12 includes the following combination of features that is not taught or otherwise suggested by the cited references:

evaluating a tag in the ML document, wherein evaluating the tag comprises reading a prefix associated with an ML namespace when the prefix is present;
detecting a definition file location attribute associated with the tag in the ML document;

fetching a definition file from a location specified by the definition file location attribute, wherein the definition file includes a list of common language

runtime namespaces, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace;

resolving the common language runtime namespace related to the tag within the definition file to establish the common language runtime class associated with the tag; and

finding an assembly that includes the common language runtime class such that the tag is mapped to the common language runtime class, wherein the assembly comprises common language runtime classes of functions associated with the common language runtime namespace.

Neither Bray nor Enns teaches or otherwise suggests "fetching a definition file from a location specified by the definition file location attribute, wherein the definition file includes a list of common language runtime namespaces, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace," "resolving the common language runtime namespace related to the tag within the definition file to establish the common language runtime class associated with the tag," and "finding an assembly that includes the common language runtime class such that the tag is mapped to the common language runtime class, wherein the assembly comprises common language runtime classes of functions associated with the common language runtime namespace." In fact, neither Bray nor Enns even mention a common language runtime namespace, a common language runtime class and/or utilizing a tag to resolve the common language runtime namespace to establish the common language runtime class. Also, neither reference teaches mapping the tag to the common language runtime class. Accordingly, applicants assert that claim 12 is allowable.

Independent claim 19 includes the following combination of features that is not taught or otherwise suggested by the cited references:

means for analyzing a tag in the ML document;

means for referencing a definition file location attribute in the ML document, wherein the definition file location attribute is related to the tag;

means for retrieving a definition file from a location specified by the definition

file location attribute, wherein the definition file includes a list of common language runtime namespaces, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace;

means for referencing *a common language runtime namespace* related to the tag within the definition file to determine *the common language runtime class* associated with the tag; and

means for locating *the common language runtime class* in an assembly such that the tag is mapped to *the common language runtime class*.

Neither Bray nor Enns teaches or otherwise suggests "means for retrieving a definition file from a location specified by the definition file location attribute, wherein the definition file includes a list of common language runtime namespaces, wherein each common language runtime namespace includes a list of common language classes associated with the common language runtime namespace," " means for referencing a common language runtime namespace related to the tag within the definition file to determine the common language runtime class associated with the tag," and " means for locating the common language runtime class in an assembly such that the tag is mapped to the common language runtime class." In fact, neither Bray nor Enns even mention a common language runtime namespace, a common language runtime class and/or utilizing a tag to reference the common language runtime namespace to determine the common language runtime class. Also, neither reference teaches mapping the tag to the common language runtime class. Accordingly, applicants assert that claim 19 is allowable.

With regard to the dependent claims, they include features that are not taught or suggested by the cited references. Furthermore, the dependent claims ultimately depend from the independent claims, respectively. As such, they should be found allowable for at least those same reasons.

III. Request for Reconsideration

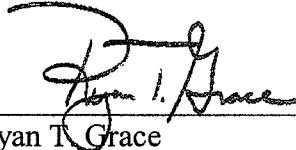
In view of the foregoing amendments and remarks, all pending claims are believed to be allowable and the application is in condition for allowance. Therefore, a Notice of Allowance is respectfully requested. Should the Examiner have any further issues regarding this application,

App. No. 10/715,709
Amendment Dated: January 24, 2008
Reply to Office Action of September 24, 2007

the Examiner is requested to contact the undersigned attorney for the applicant at the telephone number provided below.

Respectfully submitted,

MERCHANT & GOULD P.C.



Ryan T. Grace

Registration No. 52,956

Direct Dial: 206.342.6258

MERCHANT & GOULD P.C.
P. O. Box 2903
Minneapolis, Minnesota 55402-0903
206.342.6200

